Notes for summer course on robotics

Daniel Hillerström

June 20, 2013

Contents

1 Introduction

These notes are a compilation of prematerials for the academic part of the summer course on robotics programming organised by BEST Aalborg and held at University of Aalborg. A total of 21 students from a variety of European countries have been accepted for this course. The students are expected to have read these notes before the course begins. The course begins on July 12 and ends on July 22, 2013. The academic part takes place between July 14-19.

In these notes you will find a description of the course, the tools we will use and a short introduction to concepts that are essential to the course.

At `http://homes.student.aau.dk/dhille10/best/sc13` you will find the website associated with the course where all course materials are available.

2 Study regulation

This section gives a description of the course including examination and teaching form. You will also find a description of the knowledge, competences and skills which the student is supposed to have gained after course completion.

2.1 Course description

Automation of product lines for manufacturing of products and autonomous lawn mowing are only a few of many applications, for which robots are used nowadays. There is an increasing need for companies to automate processes in order to remain competitive in the consumer society of our generation. The amount of automation is not only growing in manufacturing companies, but also in places such as house holdings there is an increasing need to automate.

Robotics is a wide field of study; it ranges from software to electronic devices and further to mechanical systems. All of those individual subsets have to cooperate for the product to function. There are challenges within each of these individual parts which become even more challenging when an interaction is required. An example could be improvement of feedback signals deriving from vision, which are processed in order to use them as efficient as possible to determine the next movement.

During this course the participants will get an insight in the vast variety of challenges which follow with robotics. A deeper understanding of selected topics is gained through lectures and the knowledge is applied in a group project where a LEGO Mindstorms robot is programmed to autonomously solve a given task. The programming language used is Java, specifically the LeJOS Java implementation.

Learn about robotics, and understand the difficulties within the field. Apply knowledge gained through lectures on a practical problem and get an insight in how to apply the theory on real life applications. Compete against fellow students and show that you have the best solution to solve the predefined task.

2.2 Prerequisites

There are several prerequisites prior to the beginning of the course. Each student should fulfill the following:

- Basic knowledge of either imperative or object-oriented programming.
- Basic mathematical understanding (level of a freshman student).
- Have fair communcation skills in english including both oral and written.

Additionally each student must bring their own laptop – which should be running Linux or Windows (atleast Windows XP). Alternatively the student can use the virtual machine which is available at the course website. On their computer the following should be installed and configured prior to the course:

- Any recent version of JRE & JDK.
- Eclipse – including the LeJOS plugin.
- Subversion client

More about this in the tools section 4.

2.3 Goals

Robotics is a huge topic hence we will only cover a minor part of the subject during the course. The goal of the course is tripartite:

(i) Cover the basics of motion path-planning and handling of input data from a lightsensor.

(ii) Practical problem solving in colloboration with others.

(iii) See real-life application of robotics in the industry.

Item (i) involves both theoretical and practical understanding of core subjects in the field of robotics. The theoretical understanding is achieved through given lectures while the practical understanding, just like (ii) states, is achieved in colloboration with others. Where the students will have to work in groups on a project, in which they must solve some concrete problems, this includes: analysis of the problem, design of a solution and implementation of the solution.

For (iii) the students will visit a company which has a fully automated warehouse operated only by robots. The students will participate in a workshop where they will analyse and discuss design and implementation issues of such a system. Hence they will have to actually apply the knowledge gained from the project work.

Additionally the students will be introduced to the Problem Based Learning (PBL) model which is utilised at Aalborg University. One may find more information about the PBL model at AAU in [1].

2.4 Knowledge, skills & competences

Throughout the course the student will expand his/her knowledge in the field of robotics. After course completion the student will have acquired a specific set of skills and competences.

Knowledge

After course completion the student must have gained the following knowledge:

- Basic understanding of robotics.
- Basic understanding of motion path-planning.
- Basic knowledge of how light sources may affect sensory data from a lightsensor.
- Basic knowledge of LeJOS Java implementation & other tools used.
- Know the correct terminology.
- Know some practical applications of robotics.

Skills and competences

After course completion the student must be capable of:

- Designing and solving basic motion-based problems using LEGO Mindstorms NXT robots.
- Assessing and giving reasons for choice of methods and techniques used to solve a motion-based problem.
- Using correct terminology.
- Solving problems in colloboration with others.

2.5 Teaching method

The teaching mostly consists of project work where the participants work in groups to solve a given motion-based problem. The project work surrounds both theoretical and practical work. Theoretical work includes analysis of the problem and design of a possible solution. While practical work includes implementation and experiments.

In parallel to the project work the participants receive lectures on subjects within the field of robotics. These lectures are relevant for their project work, however the lectures present the subjects in a much more general setting. Hence the lectures give the students the theoretical foundation to help solving the problem in their project.

2.6 Examination

The examination is group-based. The exam is divided into two parts. The first part consists of a 10 minute oral presentation and demonstration of the solution. Each group member must take part in the presentation or demonstration. The second part consists of a 10 minute discussion with the examinator. Where the group members and the examinator discuss the group's proposed solution. The group members must argue for their choice of methods and techniques.

The marking of the course is "Passed/Not passed".

3 Project & lectures

This section contains a short teaser for the lectures which the student will receive during the course. Also just below in tabel 1 you will find an academic schedule for the course.

Day	Activities
Saturday	Teambuilding
Sunday	Teambuilding
Monday	Introductory lecture. Project work.
Tuesday	Recapitulation. Lecture on motion path-planning. Project work.
Wednesday	Recapitulation. Lecture on the daylight-problem. Project work.
Thursday	Recapitulation. Final preparation before the exam. Exam.
Friday	Workshop on robotics.

Table 1: Academic schedule

3.1 The project

Most of the work will be project related. The students will be working in groups of three. All groups will be given the same problem which will serve as basis for their project. In the project the group members will have to colloborate to analyse the problem domain, design a solution and implement it. It is essential to the success of the project that the group colloborates.

The problem will be a motion-based problem. Concretely, the groups must implement their solution using a LEGO Mindstorms NXT robot. The group will have to both design the robot and the software themselves. However the software must be implemented using the LeJOS Java implementation for LEGO Mindstorms. Each group will have access to one LEGO Mindstorms NXT unit.

3.2 Introductory lecture

The introductory lecture is delivered by Professor Anders Peter Ravn, whom is also the examinator. This lecture will give an introduction to the project and the programming environment. The documentation of the LeJOS Java implementation may be found in [2].

3.3 Motion path-planning

The lecture on motion path-planning is delivered by Professor Thomas Bak. This lecture will give a general introduction to the problems of motion path-planning.

Short introduction and motivation [3]: Motion planning (aka. "the navigation problem") is a term used in robotics for the process of detailing a task into discrete motions.

For example, consider navigating a mobile robot inside a building to a distant waypoint. It should execute this task while avoiding walls and not falling down stairs. A motion planning algorithm would take a description of these tasks as input, and produce the speed and turning commands sent to the robot's wheels. Motion planning algorithms might address robots with a larger number of joints (e.g., industrial manipulators), more complex tasks (e.g. manipulation of objects), different constraints (e.g., a car that can only drive forward), and uncertainty (e.g. imperfect models of the environment or robot).

3.4 Daylight-problem

The lecture on the daylight-problem is delivered by Professor Hans Jørgen Andersen. This lecture will give a general introduction to light issues regarding lightsensors.

Short introduction and motivation [4]: You "see" colors, geometry, distance, spatial continuity and discontinuity, motion, connectedness and adjacency. These notions are built into the ways in which you experience and think about the world. You interpret the information returned to your senses through a rich model of the world. The robots you build will not have such a model unless you supply it. Moreover the simple touch, light and rotation sensors that your robots will use to sense the world are very different from your tactile, visual and proprioceptive sensors and, hence, your intuitions and models may not be appropriate to apply to interpreting sensors.

3.5 Syllabus

This section contains the syllabus which gives a quick overview of the activities and estimated working hours per activity. Consult the study regulation (section 2) for information on goals, knowledge, skills and competences.

Tabel 2 contains the syllabus for the project work, while tabel 3 contains the syllabus for the lectures. Finally tabel 4 contains the syllabus for the workshop.

Activity	Project work
# working hours	≈ 24
Short summary	Practical problemsolving using LEGO Mindstorms robots.
Bibliography	Internet sources: [2] and the course website.

Table 2: Syllabus for the project work

Activity	Lectures
# working hours	≈ 5
Short summary	Gives theoretical insight into the topic of robotics.
Bibliography	None

Table 3: Syllabus for the lectures

Activity	Company visit / workshop
# working hours	≈ 6
Short summary	Workshop on designing systems operated by robots.
Bibliography	None

Table 4: Syllabus for the company visit / workshop

4 Tools

This section gives a short description of the tools we will use during the course.

4.1 Java runtime environment & development kit

Since we are going to program in Java it is essential to have a version the Java Runtime Environment (JRE) and the Java Development Kit (JDK) installed. Either any proprietary version or opensource version and preferably a relatively recent version.

An documentation for installing the proprietary versions of JDK7[1] & JRE7[2] provided by Oracle on various operating systems may be found at `http://docs.oracle.com/javase/7/docs/webnotes/install/index.html`.

If you are running Linux then it is recommended that you install the OpenJDK7 and OpenJRE7 they should both be available through the package manager.

4.2 Eclipse & the LeJOS plugin

Eclipse is an obvious choice as programming editor since we are programming in Java. However the LeJOS team provides a plugin that facilites the process of compiling, linking and uploading a program to the NXT hence we use Eclipse.

Eclipse is available via `http://www.eclipse.org/downloads/`. Again if you are running Linux then install Eclipse through your package manager. Guide on how to install and set-up the LeJOS plugin is available at `http://lejos.sourceforge.net/nxt/nxj/tutorial/Preliminaries/GettingStarted.htm`.

4.3 Revision control

Since we are working in teams it is essential to have some kind of revision control. Revision control makes it possible for a team to work concurrently on the same project without overwriting or conflicting with eachothers work. At this time of writing it has yet to be decided whether we are going to use SVN or GIT as our revision client. Since we only need basic revision control the final choice should not have any impact on the project work.

You are encouraged to install both clients. Again if you are running Linux then install them through the package manager (the SVN package

[1] Download: `http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html`

[2] Download: `http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html`

might be called Subversion). Windows users could consider TortoiseSVN available at `http://tortoisesvn.tigris.org/` as their SVN client and the Git client available at `http://git-scm.com/downloads`.

4.4 The virtual machine

If you struggle with installing and configuring the tools – fear not! We will provide a fully configured lightweight virtual machine that works out-of-the-box with the all tools that we will be using. This virtual machine will be available at the course website before the beginning of the course. It will be a VirtualBox virtual machine and hence you are required to install VirtualBox to be able to run it. VirtualBox is available at `https://www.virtualbox.org/wiki/Downloads`, you may also find the documentation there.

References

[1] PBL Academy. *The PBL Academy at Aalborg University.* `http://www.pbl.aau.dk/`, 2013. 4

[2] LeJOS Team. *LeJOS, Java for Lego Mindstorms.* `http://lejos.sourceforge.net`, 2013. 6, 7

[3] Wikipedia community. *Motion planning.* `https://en.wikipedia.org/wiki/Motion_planning`, 2013. 6

[4] Tom Dean. *Sensors and Sensing.* `http://cs.brown.edu/~tld/courses/cs148/02/sensors.html`, 2002. 7